

FieldView では、FVX によるバッチ処理を行うことができますが、
ここでは、CFD 解析の計算ループ中に 画像出力する事例をご案内いたします。

* 事例には OpenFOAM を利用しております。

HT032_sample.pw

Pointwise を利用した メッシュデータサンプルです。

以下のディレクトリ内には、すでに polyMesh データは含まれています。

HT032/

OpenFOAM の計算事例ディレクトリ(ケースディレクトリ)です。

constant/	メッシュなどのディレクトリ(今回は関係なし)
orig0/	初期条件を入れたディレクトリ(今回は関係なし)
origMesh/	作業したメッシュデータを格納するディレクトリ(今回は関係なし)
system/	計算メソッド、計算条件類 ディレクトリ(今回は関係なし)
orgCreateImage.fvx	サンプル FVX
transient_fv.sh	サンプル FVX を一定時刻置きに実行するシェルスクリプト

《使い方》 ケースディレクトリ内に移動し、以下のコマンドを順次入力します

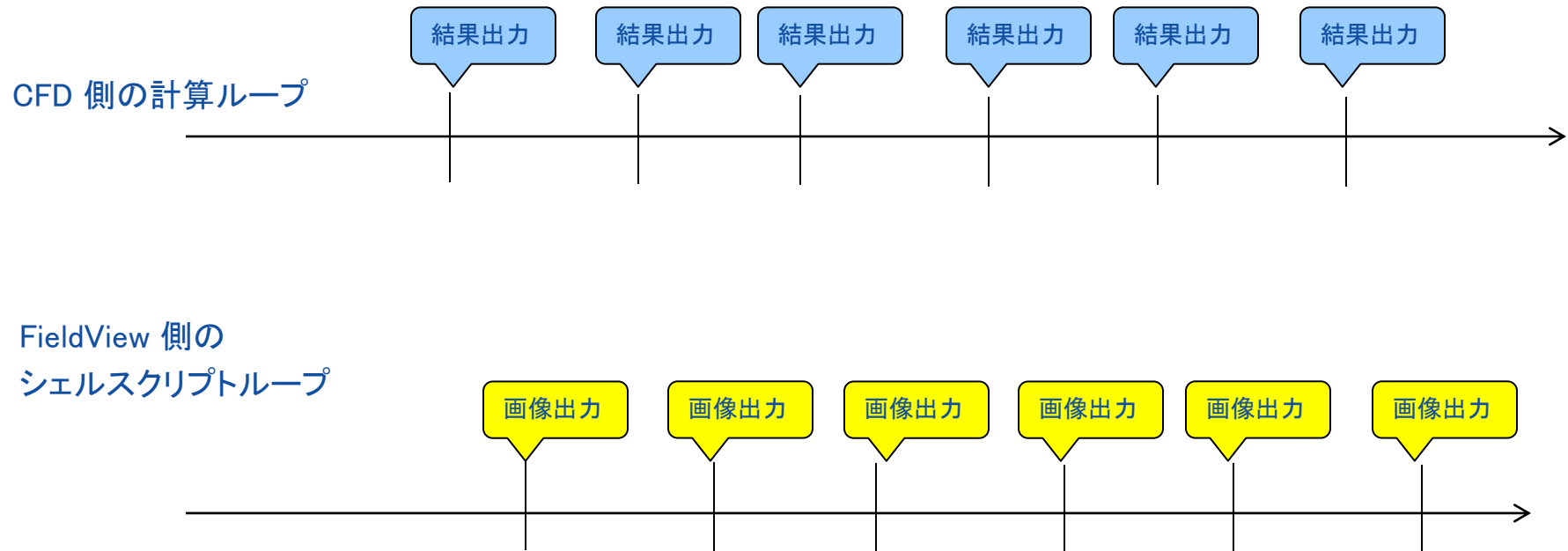
```
make clean    [enter]    ... 前に流した計算データをクリアする
make init     [enter]    ... origMesh 内のメッシュデータを constant/polyMesh へコピー
make fgrun    [enter]    ... Makefile 内で指定された ソルバを実行
               * 本 サンプルでは buoyantBoussinesqPimpleFoam を利用(シングルコア)
```

計算が流れてることを確認したら、
別のコンソールウィンドウで、同じディレクトリへ移動し、

```
./transient.sh [enter]
```

あるいは

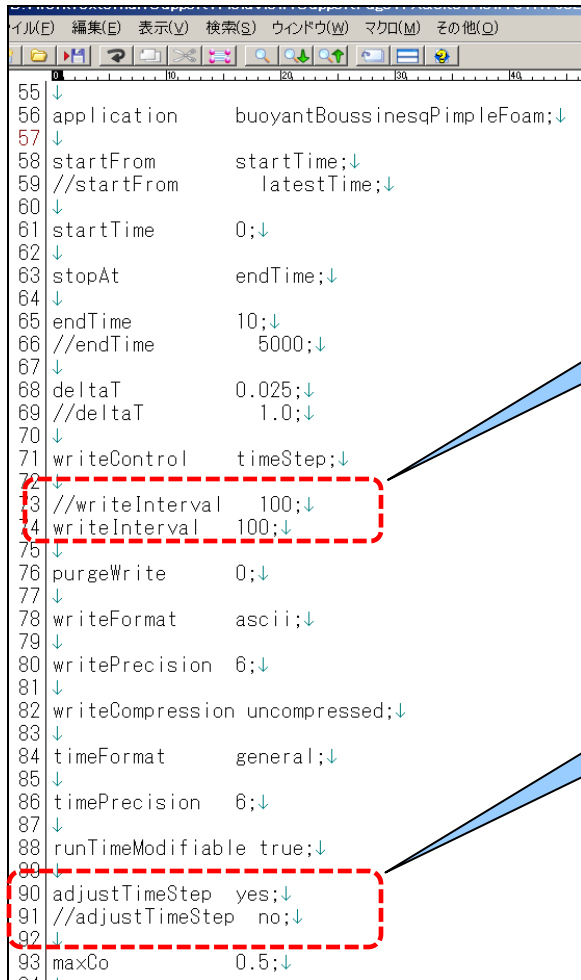
```
source ./transient_fv.sh [enter]
```



今回のサンプルシェルスクリプトは CFD のジョブとは別個に稼働します。
上記のように、タイミングよくシェルスクリプトが稼働するように パラメータを設定する必要があります。

ソルバ側のデータ出力タイミングに関わる箇所

system/ControlDict ファイル内



```
55 ↓
56 application      buoyantBoussinesqPimpleFoam; ↓
57 ↓
58 startFrom        startTime; ↓
59 //startFrom      latestTime; ↓
60 ↓
61 startTime        0; ↓
62 ↓
63 stopAt           endTime; ↓
64 ↓
65 endTime          10; ↓
66 //endTime        5000; ↓
67 ↓
68 deltaT           0.025; ↓
69 //deltaT         1.0; ↓
70 ↓
71 writeControl      timeStep; ↓
72 ↓
73 //writeInterval   100; ↓
74 writeInterval     100; ↓
75 ↓
76 purgeWrite       0; ↓
77 ↓
78 writeFormat       ascii; ↓
79 ↓
80 writePrecision    6; ↓
81 ↓
82 writeCompression  uncompressed; ↓
83 ↓
84 timeFormat        general; ↓
85 ↓
86 timePrecision     6; ↓
87 ↓
88 runTimeModifiable true; ↓
89 ↓
90 adjustTimeStep    yes; ↓
91 //adjustTimeStep  no; ↓
92 ↓
93 maxCo            0.5; ↓
94 ↓
```

結果出力の
イタレーションインターバル

Courant 数を自動調整する機能を on
時間刻みが変わります

シェルスクリプト側の画像出力タイミングに関わる箇所 transient_fv.sh ファイル内

The image shows a terminal window displaying the contents of the `transient_fv.sh` script. The script is a shell script that runs a loop of FVX (FieldView) processing. Annotations with callout boxes highlight specific parts of the script:

- FieldView 実行ファイルパス**: Points to the line `fv_exec="/usr/local/FieldView/fv13/fv/bin/fv"`.
- キックする FVX ファイル**: Points to the line `FileFVX="/orgCreateImage.fvx"`.
- 一時停止する 秒数**: Points to the line `SleepTime="10"`, which is highlighted with a red dashed box.
- 最後の時刻だけ FV-UNS コンバート**: Points to the lines `### FV-UNS convert only latest time` and `foamToFieldview9 -latestTime`, which are also highlighted with a red dashed box.

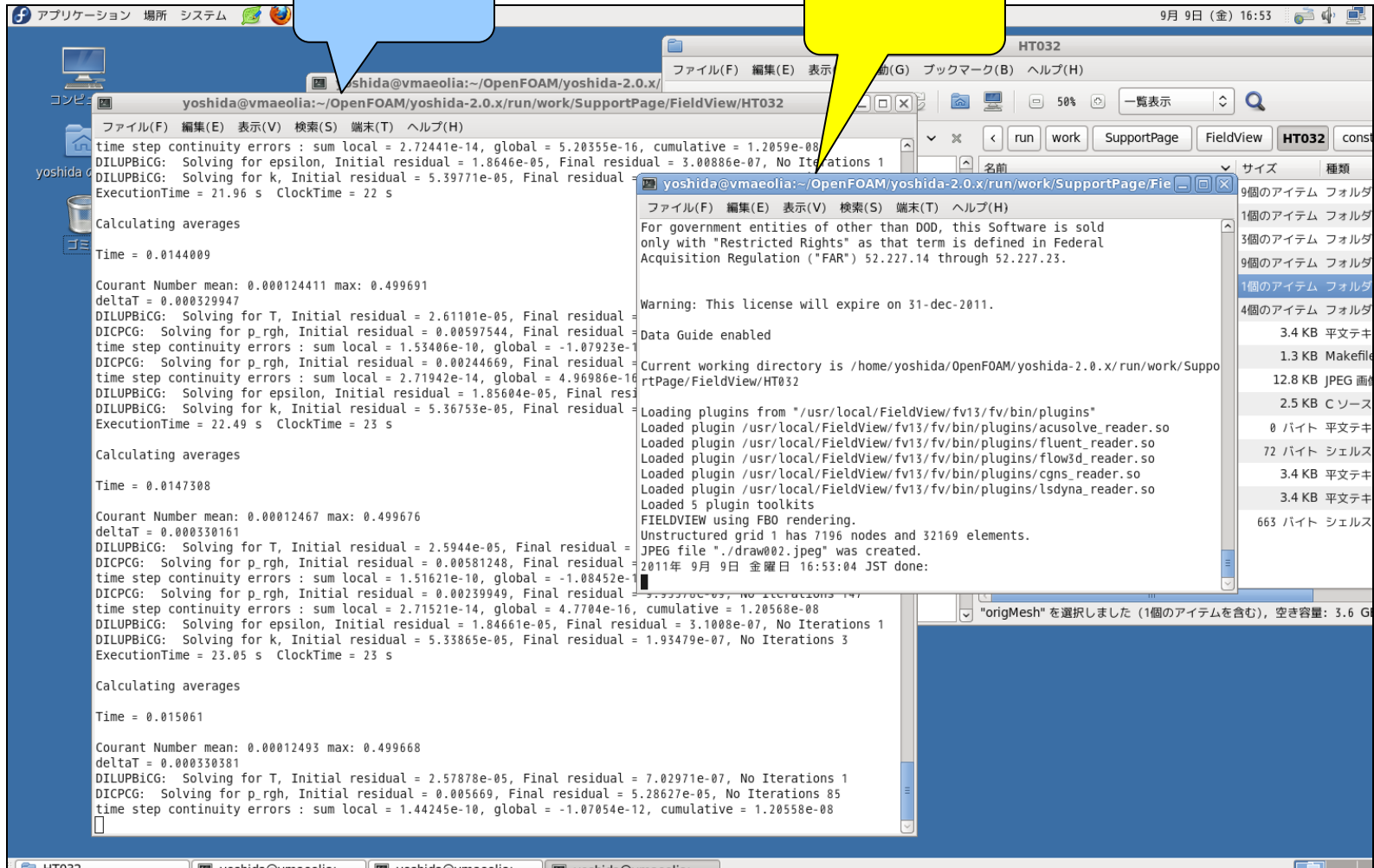
```

1  #!/bin/bash
2
3  fv_exec="/usr/local/FieldView/fv13/fv/bin/fv"
4  FileFVX="/orgCreateImage.fvx"
5  SleepTime="10"
6
7
8  #make fgrun
9
10 nCount=1
11 while true; do
12
13     nCount=$((nCount+1))
14     echo "nCount : $nCount"
15
16     ### FV-UNS convert only latest time
17     foamToFieldview9 -latestTime
18
19     ### detect latest FV-UNS file name
20     LatestUnsFile=`ls -la ./Fieldview/*.uns | head -n1 | awk '{printf($8, "%c")}'`
21     echo "Latest FV-UNS: ${LatestUnsFile}"
22
23     sed "s@strFileName@${LatestUnsFile}@g" ${FileFVX} > ./tmp.fvx
24     sed "s@numF@${nCount}@g" ./tmp.fvx > ./CreateImage.fvx
25
26     ### execute FVX batch
27     $fv_exec -batch -fvx ./CreateImage.fvx
28
29     echo `date` "done:"
30     sleep ${SleepTime}
31
32 done
  
```

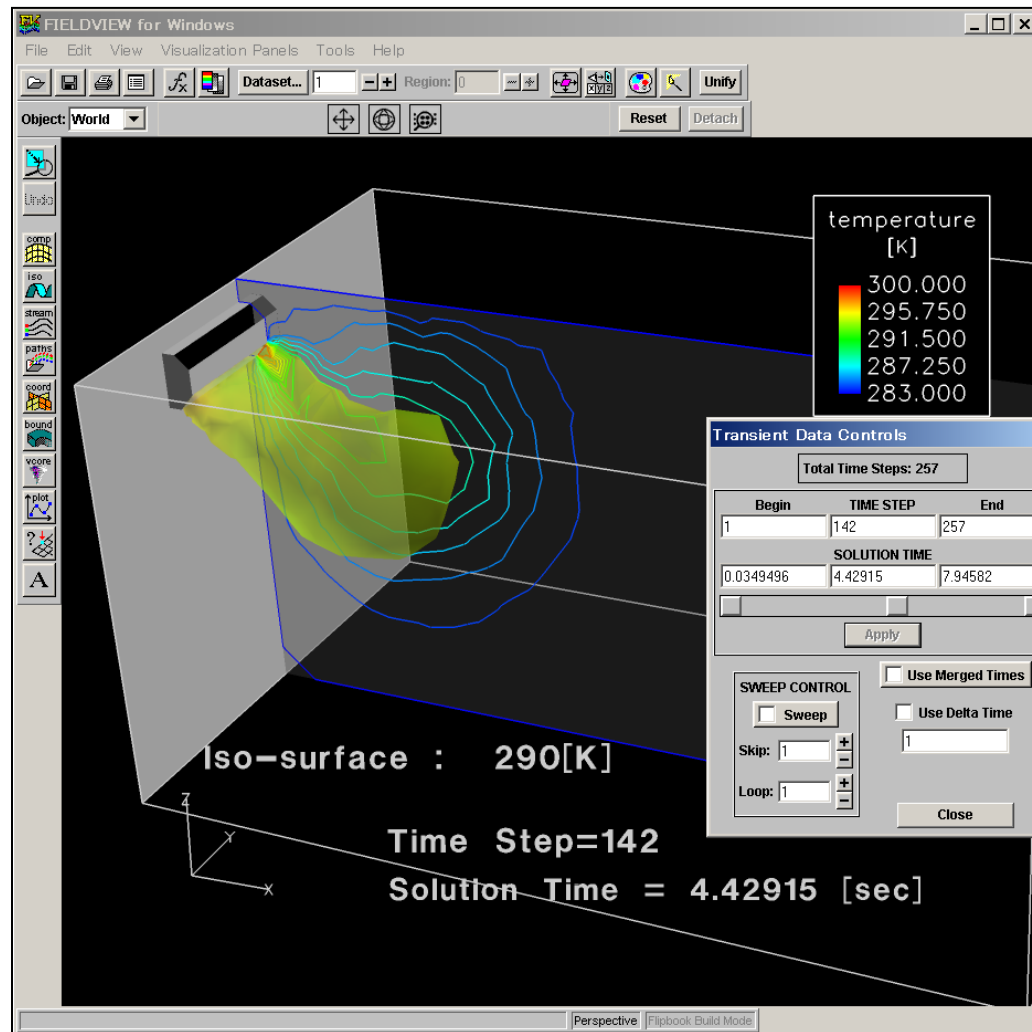
処理風景

CFD 側の loop

FVX 側の loop



解析事例: エアコンを模擬した 空調計算事例を使っています。



サンプルで
実際に出力し簡易画像

